Criteria weights assessment through prioritizations (WAP) software tool

Athanasios Spyridakos, Nikos Tsotsolas, Isaak Vryzidis
University of West Attica

Structure

- Robustness Concern in Criteria Weights Inference
- Robust Simos Method Approach
- Robustness Analysis Tools and Feedbacks
- WAP Tool
- What’s next

Robustness Concern in Criteria Weights Inference

To assess the criteria importance weights for non-compensatory MCDA models like the outranking methods (ELECTRE, PROMETHEE, …) or some multiobjective optimization methods (goal programming, compromise programming, …) we use two kinds of methods:

i. direct assessment procedures, where the DM is asked to explicitly express the criteria weights in terms of percentages, and

ii. indirect methods aiming to infer the weights from pairwise comparisons of the criteria or reference alternatives. (eg.Simos Cards Method).

Simos Method

Collecting the information:

- STEP 1: We give to the DM **a set of cards** with criterion name on it. We also give **a set of white cards** with the same size.

- STEP 2: We **ask the DM to rank these cards from the least important to the most important**. If some criteria have the same importance (i.e., the same weight), DM should build a subset of cards holding them together with a clip. Consequently, we obtain a complete pre-order on the whole of the *n* criteria.

- STEP 3: We ask the DM to **introduce white cards** in between of two successive subsets of criteria **for increasing the difference of importance** of criteria. The greater the difference between the mentioned weights of the criteria (or the subsets), the greater the number of white cards.

Simos Algorithm

The information provided by the DM is utilized by the Simos method for the determination of the weights, according to the following algorithm:

i. ranking of the subsets of ex aequo from the least important to the most important, considering also the white cards,

ii. assignment of a position to each criterion/card and to each white card,

iii. calculation of the non-normalized weights, and

iv. determination of the normalized weights.

Simos Method – Pros + Cons

Pros

- Simplicity → Convenient for the DM to express his/her preference

- Easy calculations

Cons

- Robustness issues:
  - arbitrarily calculation of a **unique weighting vector**
  - even though there exist infinitely more weight vectors, also satisfying DM's preferential statements

Simos Method – LP Model

For every $j = 1, 2, \ldots, n-1$, and $h = 1, 2, \ldots, k$:

- If g_j is followed by g_{j+1}, and g_{j+1} belongs to the same importance class as g_j, write:

$$p_j = p_{j+1}$$

- If g_j is followed by g_{j+1}, and g_{j+1} belongs to a most importance class, write:

$$p_j < p_{j+1} \quad \Leftrightarrow \quad p_{j+1} - p_j \geq \delta$$

- If between g_j and g_{j+1} a white card wc_h is placed, write:

$$p_j < w_h \;\; \text{and} \;\; w_h < p_{j+1} \quad \Leftrightarrow \quad w_h - p_j \geq \delta \;\; \text{and} \;\; p_{j+1} - w_h \geq \delta$$

- $p_1 + p_2 + \ldots + p_n = 1$

- $p_1 \geq 0, p_2 \geq 0, \ldots, p_n \geq 0; w_1 \geq 0, w_2 \geq 0, \ldots, w_k \geq 0$

- In case of the revised Simos method, the following equation should be added:

$$p_n = zp_1$$

Then all the Simos weighting solutions belong to the polyhedral set:

$\boldsymbol{P} = \{ \boldsymbol{p} \in R^n \; / \boldsymbol{p}$ satisfies the system of these linear relations

Robust Simos Method – Robustness Rules

1. **Compute the ranges of variation** of each separate criterion, by solving *2n* (where n the number of criteria) linear programs of the following type (MAX-MIN approach): *Min* p_j *& Max* p_j, for $j = 1, 2, \ldots, n$ *s.t.* $\boldsymbol{p} \in \boldsymbol{P}$
2. **Compute the average weighting vector** ("barycenter") of all different vectors (from the 2n solutions obtained in the former rule) - representative solution.
3. **Find and record all the vertices of the polyhedron** $\boldsymbol{P}$, by using the Manas-Nedoma (1968) analytical algorithm
4. **Implement a random weight sampling** algorithm/technique to produce and statistically analyze a great number of weighting sets from the polyhedron.
5. **Visualize the ranges of variation** of the criteria weights and/or the polyhedron they define, in order to perceive the extent of the instability.
6. **Calculate the ratio of the volume of the criteria polyhedron by** following the implementation of recommendation 4, and the unconstrained criteria area.
7. **Compute the robustness measure ASI** (Average Stability Index), which is the mean value of the normalized standard deviation of the estimated weights:

$$\mathrm{ASI} = 1 - \frac{1}{n}\sum_{i=1}^{n} \frac{\sqrt{m\sum_{j=1}^{m} p_{ij}^2 - \left(\sum_{j=1}^{m} p_{ij}\right)^2}}{\frac{m}{n}\sqrt{n-1}}$$

Robust Simos Method – Robust decision aiding

1. **Built** on *A* two distinct outranking relations, the **necessary outranking** (aS^Nb ⇔ *aSb, i.e. action a outranks action b, for every weighting vector* $\boldsymbol{p} \in \boldsymbol{P}$), and the **possible outranking** (aS^Pb ⇔ *there is at least one weighting vector* $\boldsymbol{p} \in \boldsymbol{P}$ *for which aSb*; (see Figueira et al. (2009) (Greco et al. 2008) for definitions and properties of these outranking relations.

2. **Define the maximum and minimum possible ranking positions for every action** in *A* with mixed integer linear programming techniques (see Kadzinski et al., 2012).

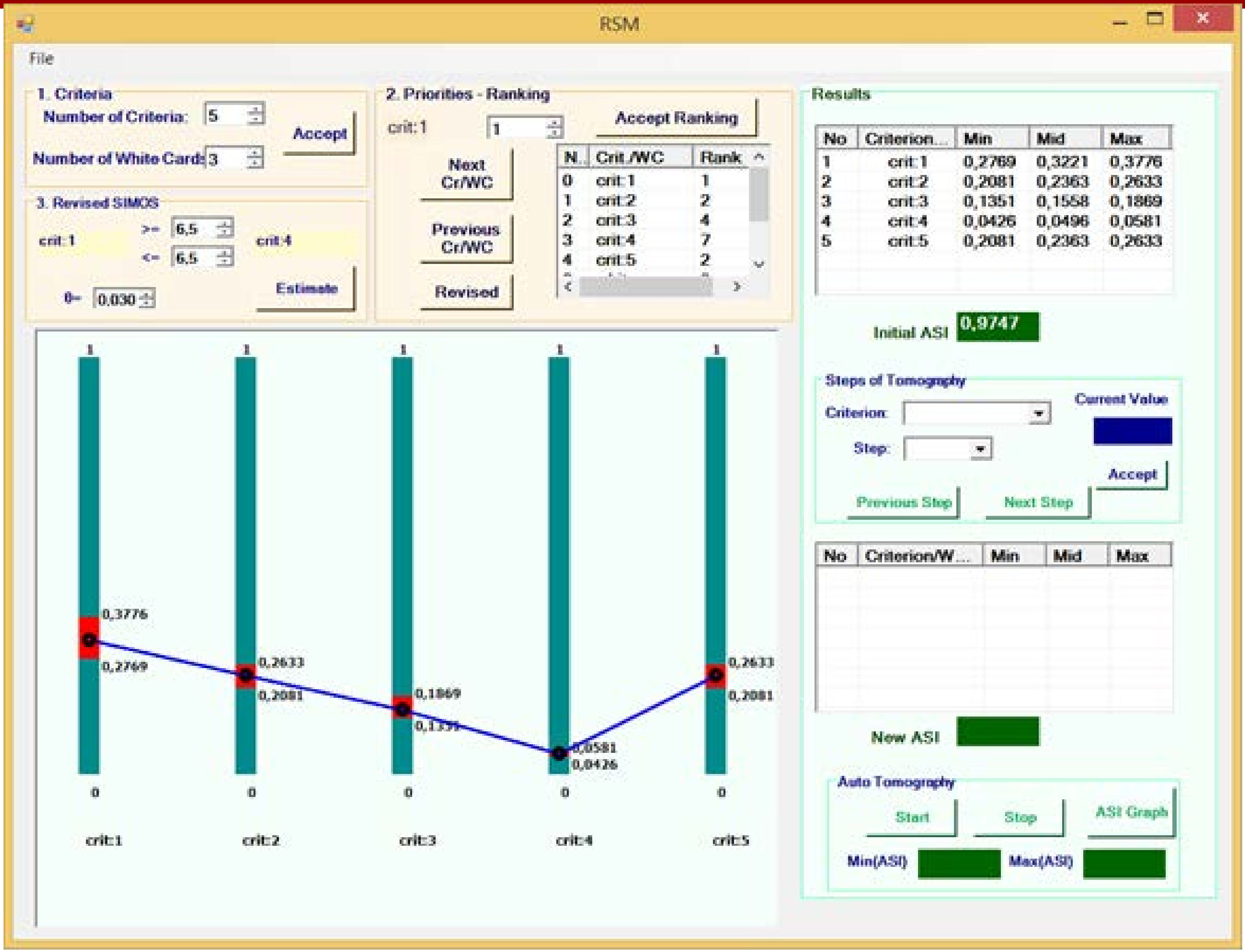
3. **Statistically compute the possibility/probability that an action *a* belongs to the kernel of the outranking graph** in the cases of ELECTRE I and ELECTRE IS choice methods,.

4. **Following a random sampling** in $\boldsymbol{P}$, **compute entropy measures** associated to outranking relations between actions in *A* and ranking positions for each action separately (see Greco et al., 2013).

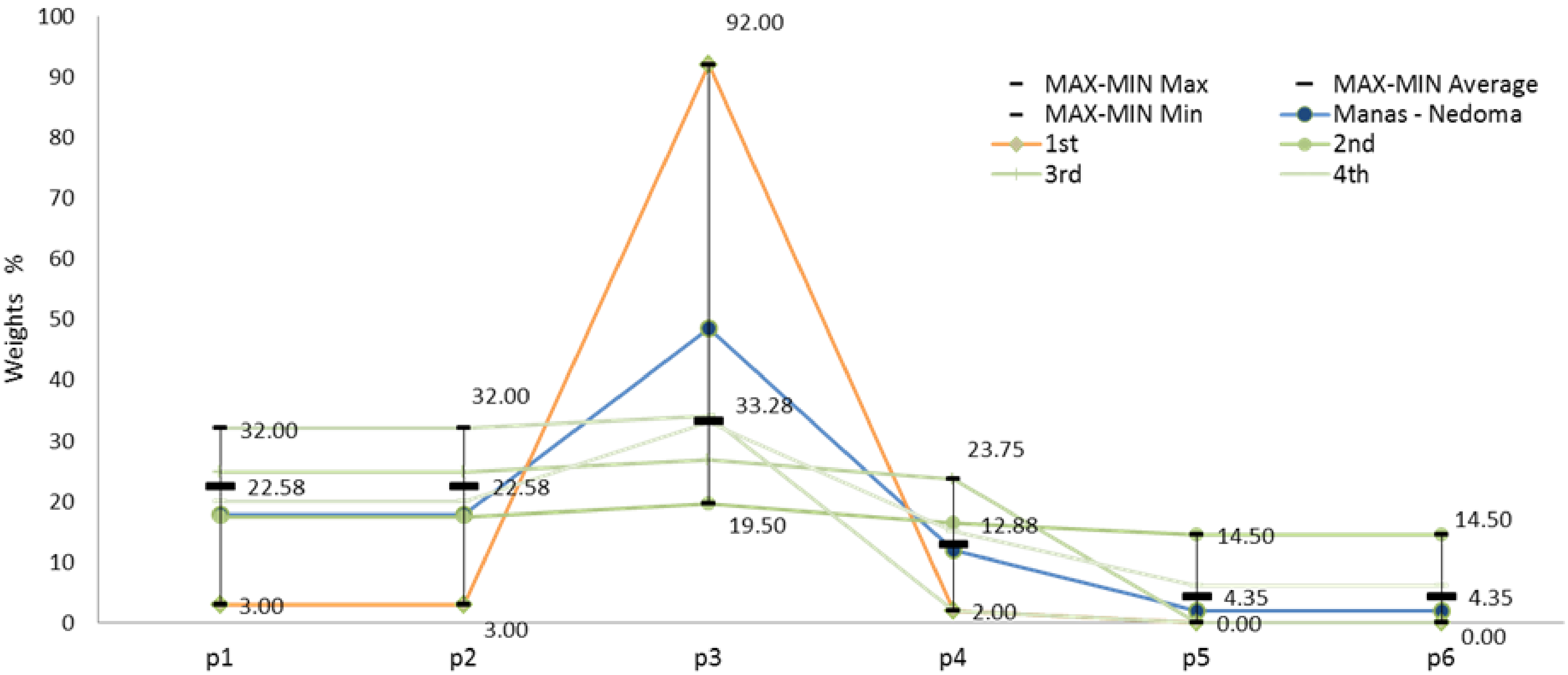
Suite of RA Tools (1)

Robust Simos
(RS)

Robust Simos Method – SW Tool

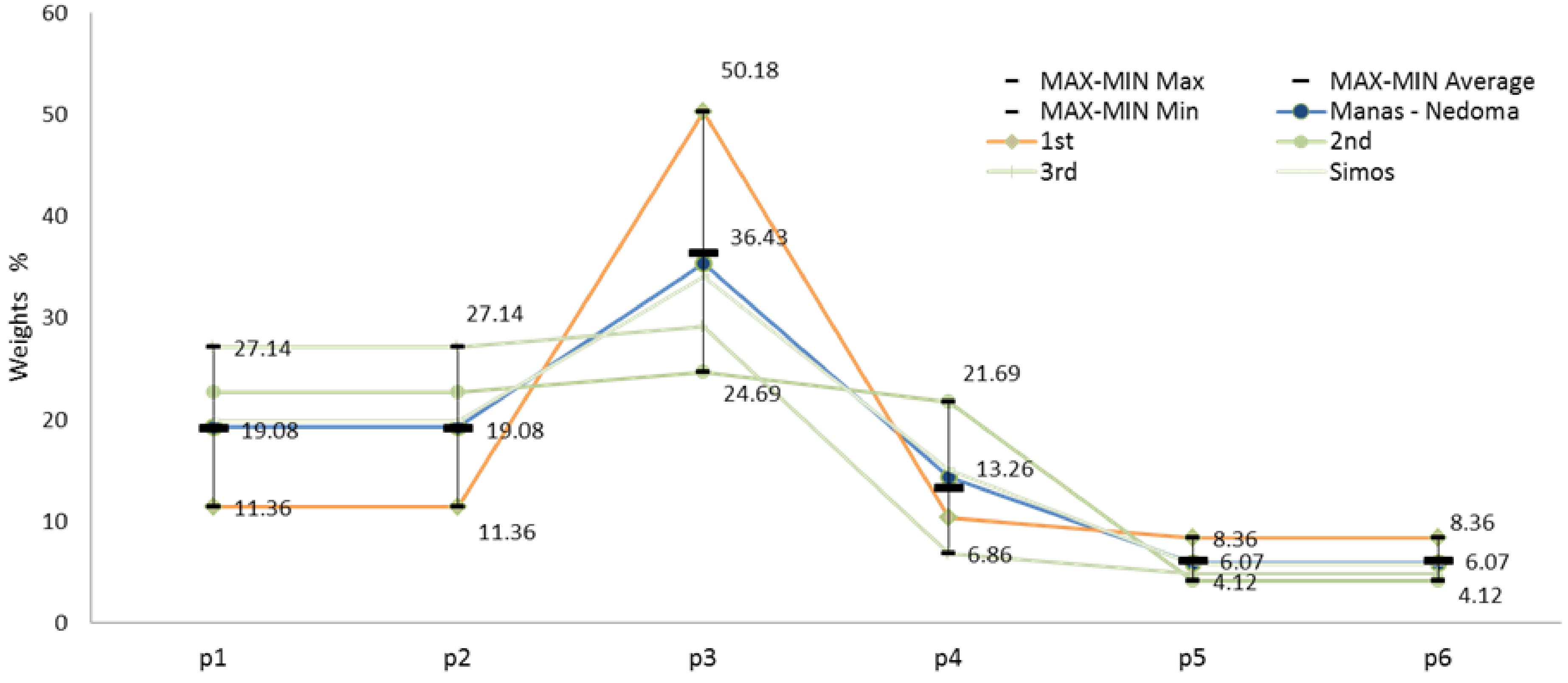


Robust Simos Method – Example (1)



The calculated range of the weights using different algorithms (2 white cards)

Robust Simos Method – Example (2)



The calculated range of the weights using different algorithms (2 white cards) (Revised Simos)

Robust Simos Method – Example (3)

Ranking of 6 alternatives using ELECTRE II with PROMETHEE II sorting procedure (case with 2 white cards).

	1st	2nd	3rd	4th	5th	6th
Simos	B,D	–	C	A,F	–	E
Max–Min average	B	D	C	A,F	–	E
Manas–Nedoma average	B,C,D	–	–	A	F	E
1st vertex of the polyhedron	C	D	B	A	F	E
2nd vertex of the polyhedron	B,D	–	C	A,F	–	E
3rd vertex of the polyhedron	B,D	–	C	A,F	–	E
4th vertex of the polyhedron	E	F	A	B	D	C

Ranking of 6 alternatives using ELECTRE II with PROMETHEE II sorting procedure (case with 2 white cards and **revised Simos**).

	1st	2nd	3rd	4th	5th	6th
Revised Simos	B,D	–	C	A,F	–	E
Max–Min average	B,D	–	C	A,F	–	E
Manas–Nedoma average	B,D	–	C	A,F	–	E
1st vertex of the polyhedron	C,D	–	B	A,F	–	E
2nd vertex of the polyhedron	B,D	–	C	A,F	–	E
3rd vertex of the polyhedron	E	F	A	B	D	C

Robust Simos Method – Example (4)

Extreme ranking positions of alternatives (case with 2 white cards)

	A	B	C	D	E	F
$P^*(a)$	3rd	1st	1st	1st	1st	2nd
$P_*(a)$	4th	4th	6th	5th	6th	5th

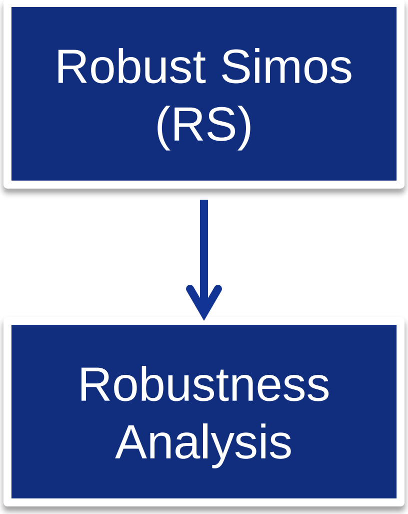
Extreme ranking positions of alternatives (case with 2 white cards 7 **revised Simos**)

	A	B	C	D	E	F
$P^*(a)$	3rd	1st	1st	1st	1st	2nd
$P_*(a)$	4th	4th	6th	5th	6th	4th

ASI of the different algorithms

	No white card		2 white cards			
	Manas–Nedoma	Max–Min	Manas–Nedoma (no z)	Max–Min (no z)	Manas–Nedoma ($z = 6$)	Max–Min ($z = 6$)
ASI	0.662	0.659	0.672	0.677	0.844	0.842

Suite of RA Tools (2)



Robustness Analysis Tools and Feedbacks

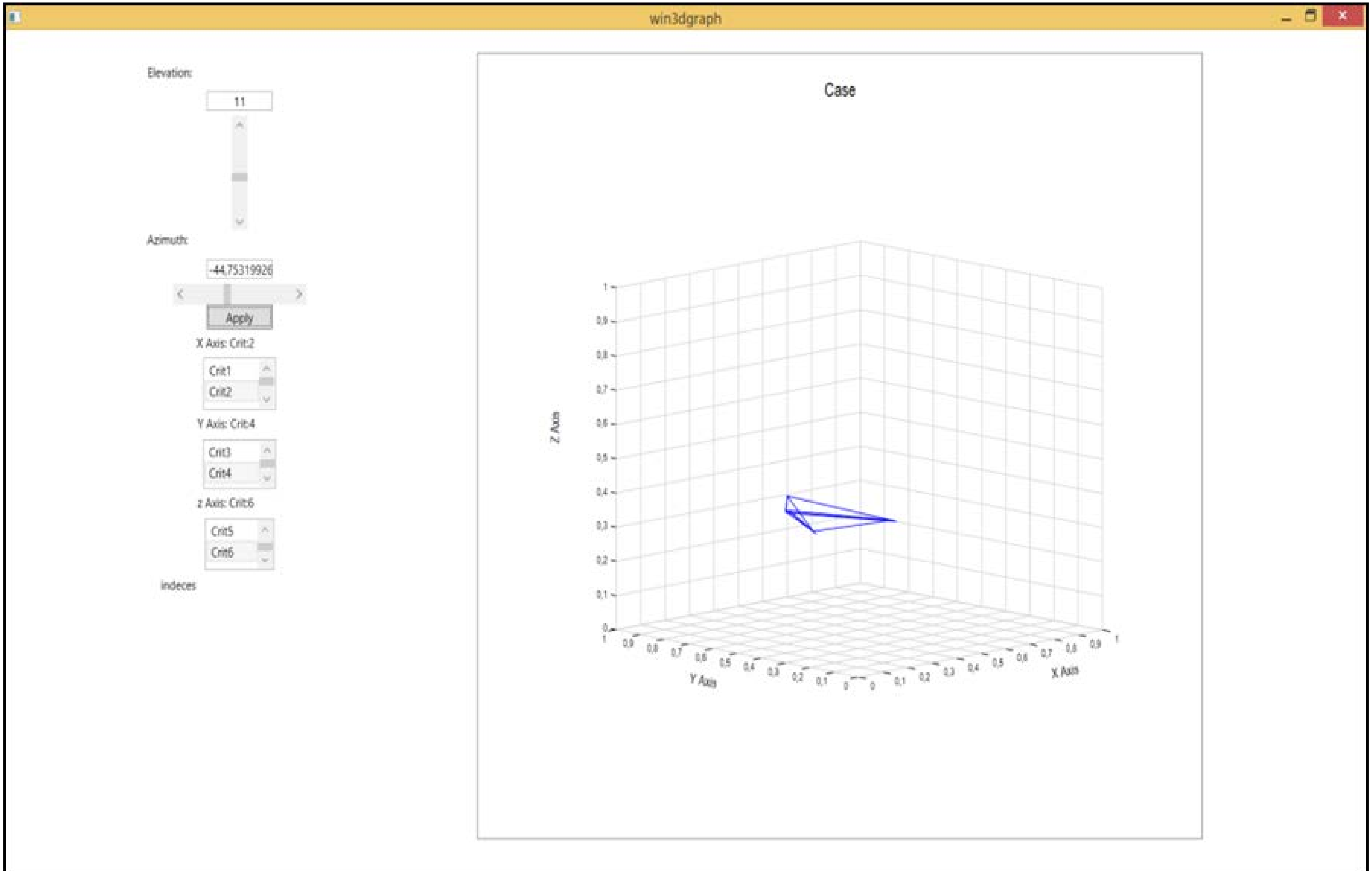
- Visualisation of the hyper-polyhedron (**robustness rules 1 & 5**)

- Measurement of the robustness (**robustness rule 1 & 7**)

- Tomographical analysis (**robustness rule 1 & 4**)

Visualisation of the hyper-polyhedron

Visualisation of the hyper-polyhedron in a 3-D graphical interface so as to provide a view of the solution's hyper-space for the selected 3 dimensions.



Measurement of the robustness

A set of indices and special data handling features utilised and designed in order to provide a clear and precise view of the robustness level.

$$\mu_i = (\max(p_{ij}) - \min(p_{ij})),\ \text{where}$$

p_{ij} the weight of the i criterion in the j vertex, resulted by post optimal analysis,

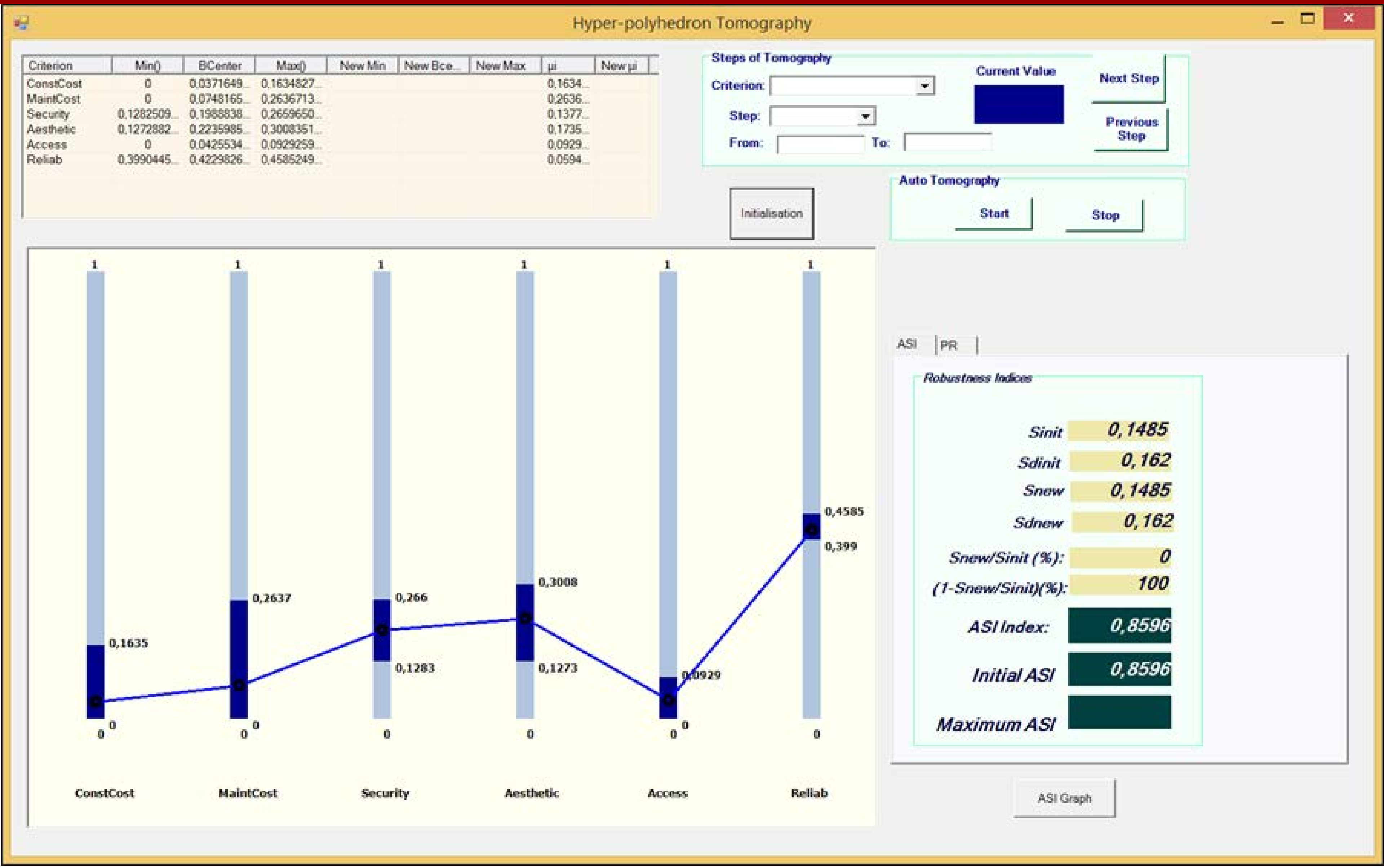
$i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, m,\ $ n : number of criteria and m : number of hyper - polyhedron vertices

$$S = \frac{\sum_{i=1}^{n} \mu_i}{n}, \qquad S_d = \sqrt{\frac{\sum_{i=1}^{n} (\mu_i - S)^2}{n - 1}},\quad \text{n : number of criteria}$$

$$ASI = 1 - \frac{\sum_{i=1}^{n} \sqrt{\left(m\left(\sum_{j=1}^{m} (p_{ij})^2 \right) - \left(\sum_{j=1}^{m} p_{ij} \right)^2 \right)}}{m\sqrt{(n-1)}},$$

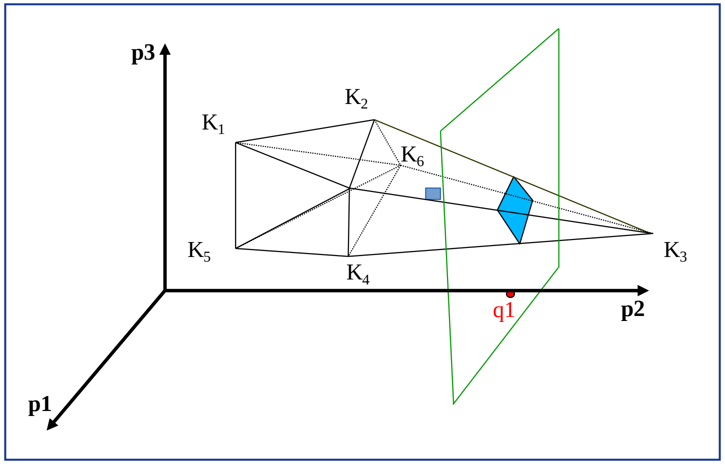
where n : number of criteria and m : number of hyper - polyhedron vertices

Measurement of the robustness



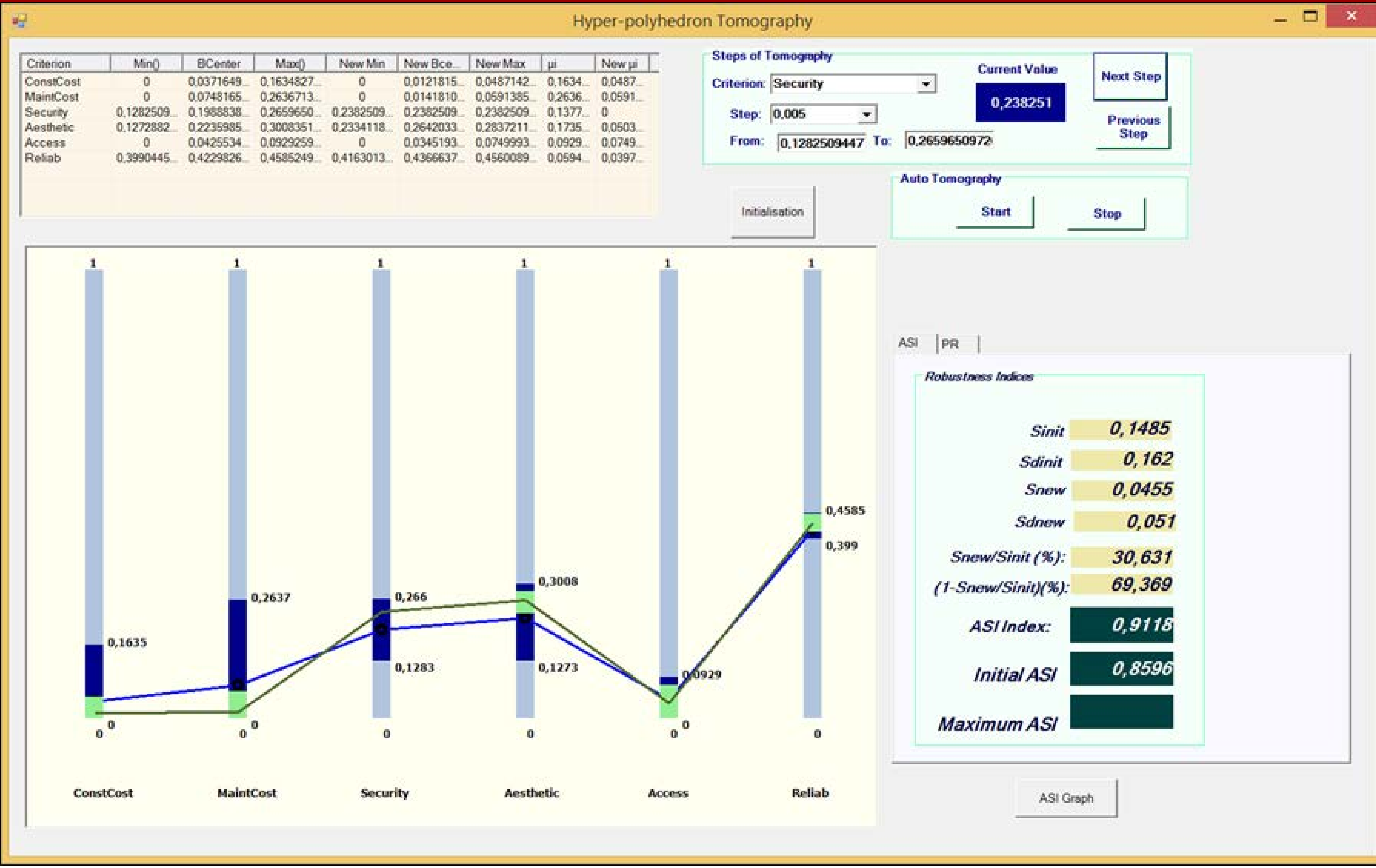
Tomographical analysis

Tomographical approach constitutes a way to picture the degree of robustness into the hyper-polyhedron. Discretisation approach by using n-1 dimensional cutting hyper-planes, creating half-spaces in the n-dimensional space estimated hyper-polyhedron of the criteria weights.



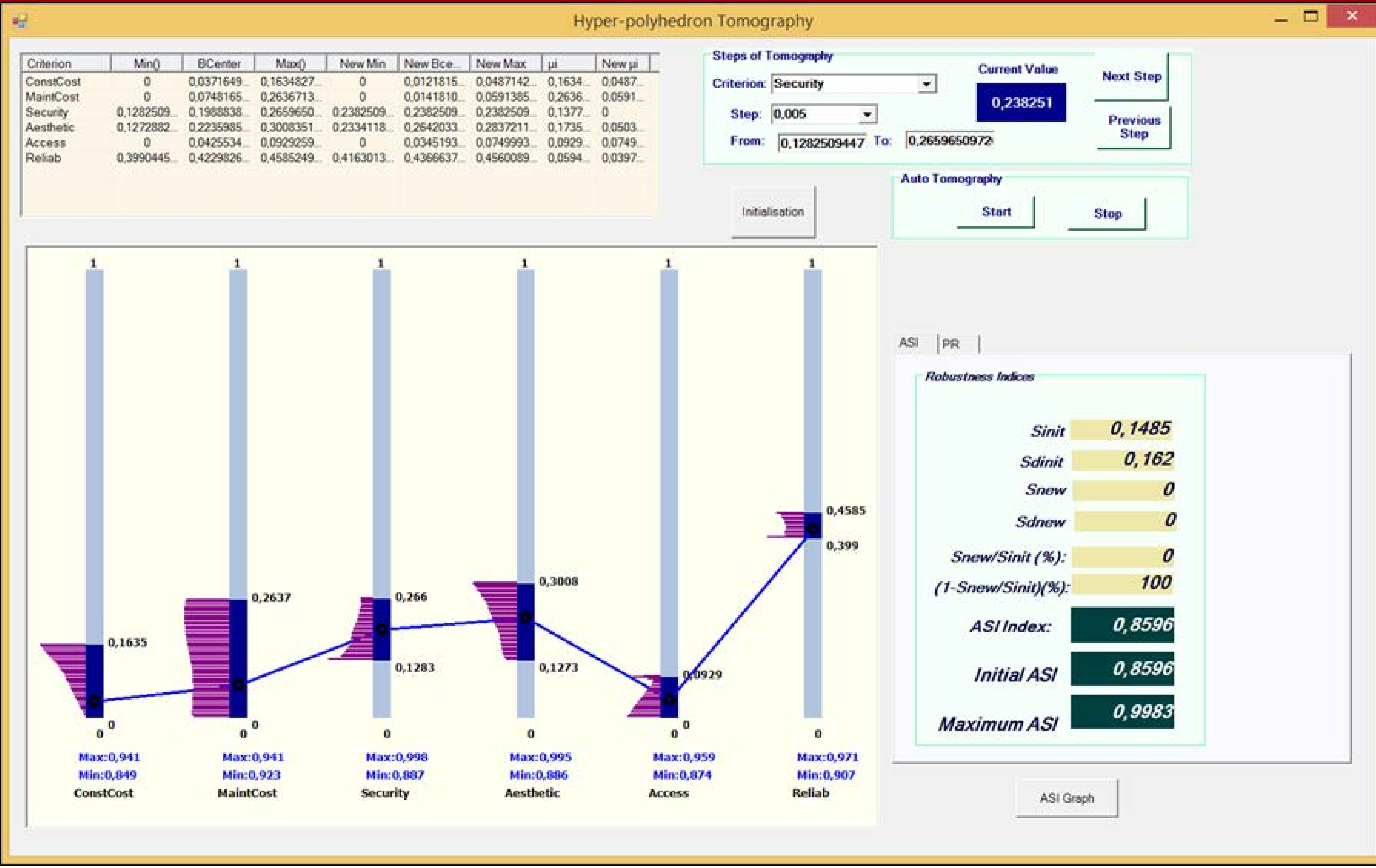
a. **Manually inspection** of the robustness of the hyper-polyhedron by selecting a criterion and a step t, for the discretisation, from a list of predefined values (0.001, 0.005, ..., 0.1).
b. **Automatic running of the topography** for all the criteria with a pre-selected step t, for the construction of cutting hyper-planes and calculation of the corresponding indices for the robustness evaluation and the presentation of the results.

Tomographical analysis (manually)

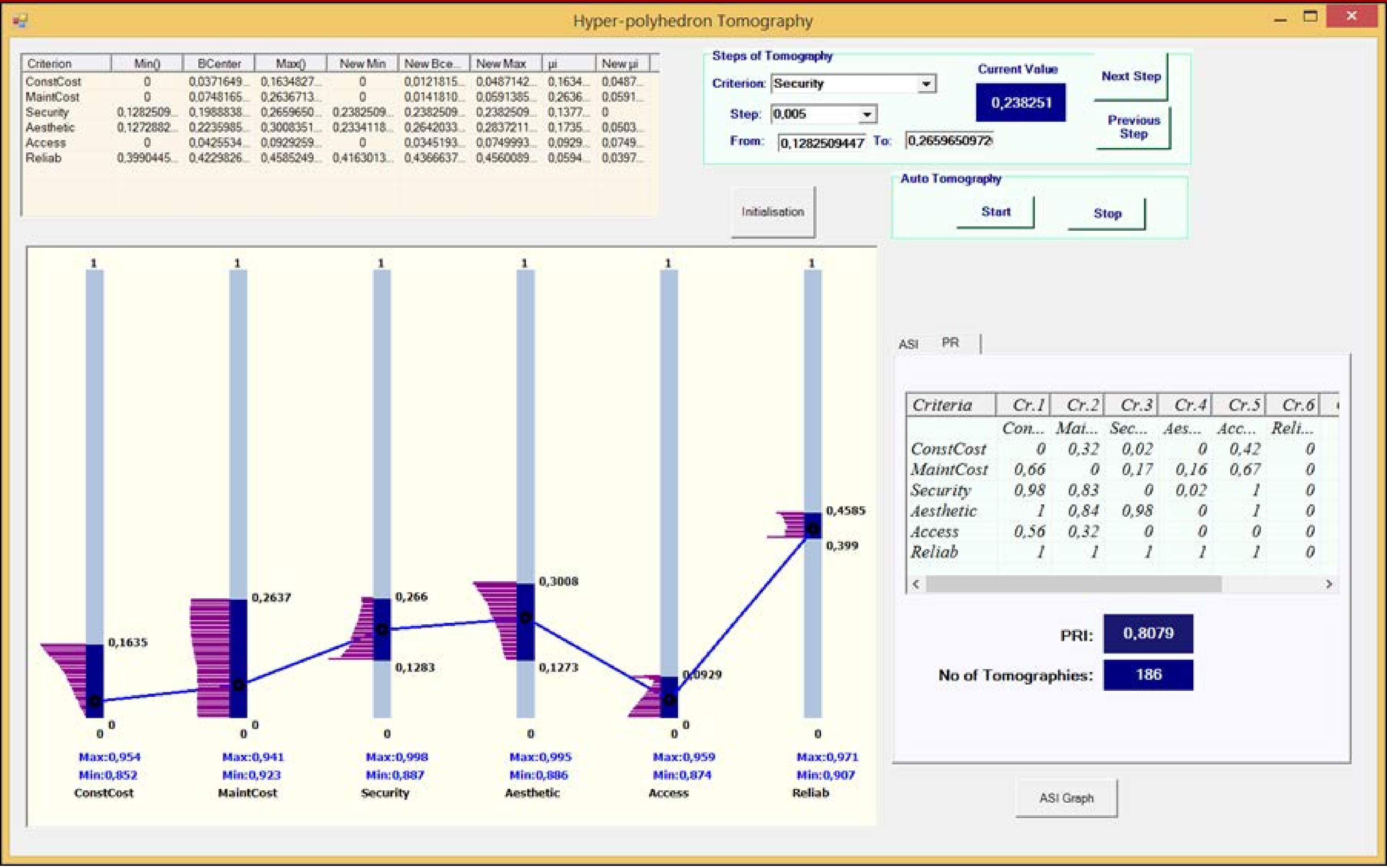


Criterion	Min()	BCenter	Max()	New Min	New Bce...	New Max	µi	New µi
ConstCost	0	0,0371649...	0,1634827...	0	0,0121815...	0,0487142...	0,1634...	0,0487...
MaintCost	0	0,0748165...	0,2636713...	0	0,0141810...	0,0591385...	0,2636...	0,0591...
Security	0,1282509...	0,1988838...	0,2659650...	0,2382509...	0,2382509...	0,2382509...	0,1377...	0
Aesthetic	0,1272882...	0,2235985...	0,3008351...	0,2334118...	0,2642033...	0,2837211...	0,1735...	0,0503...
Access	0	0,0425534...	0,0929259...	0	0,0345193...	0,0749993...	0,0929...	0,0749...
Reliab	0,3990445...	0,4229826...	0,4585249...	0,4163013...	0,4366637...	0,4560089...	0,0594...	0,0397...

Tomographical analysis (auto-ASI)

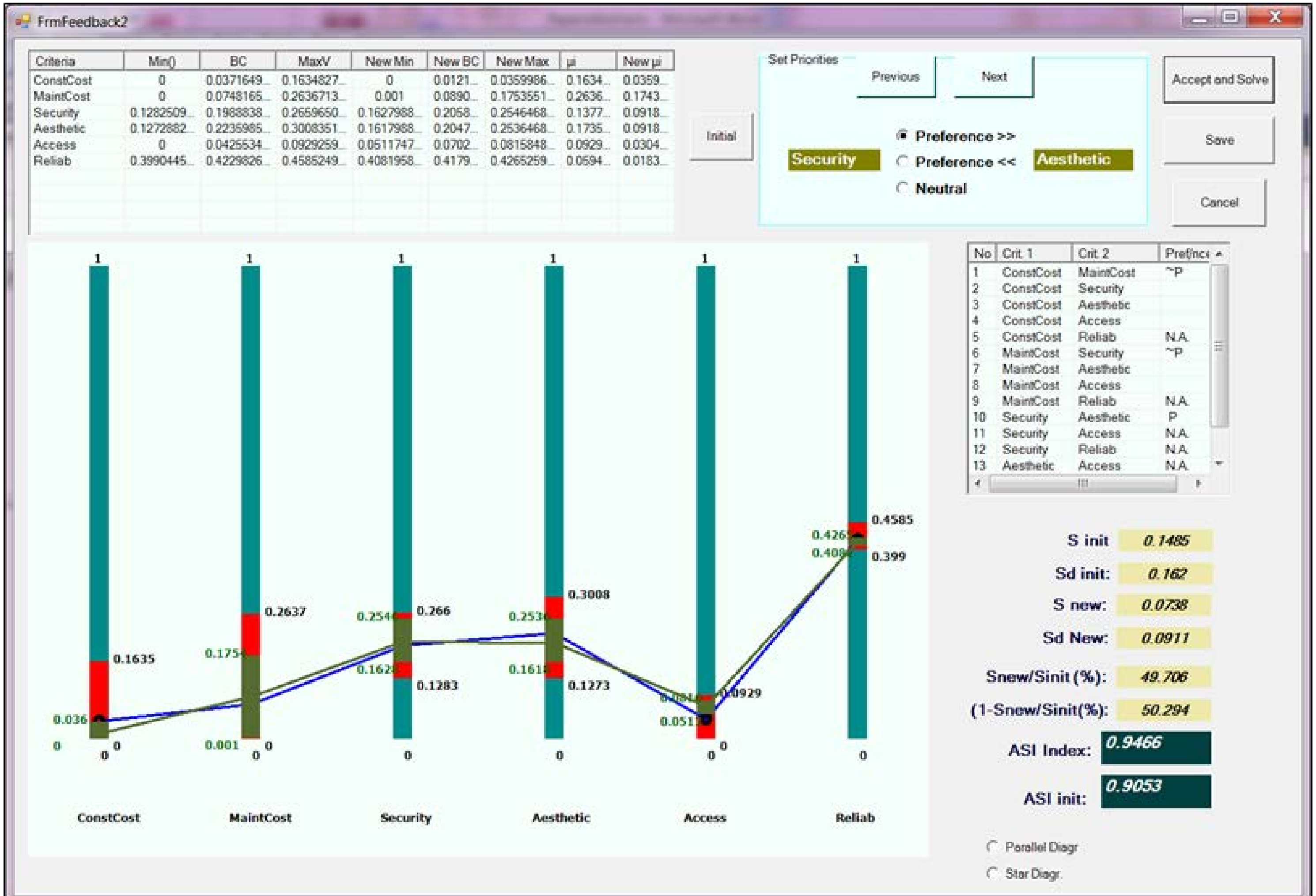


Tomographical analysis (auto-PR)



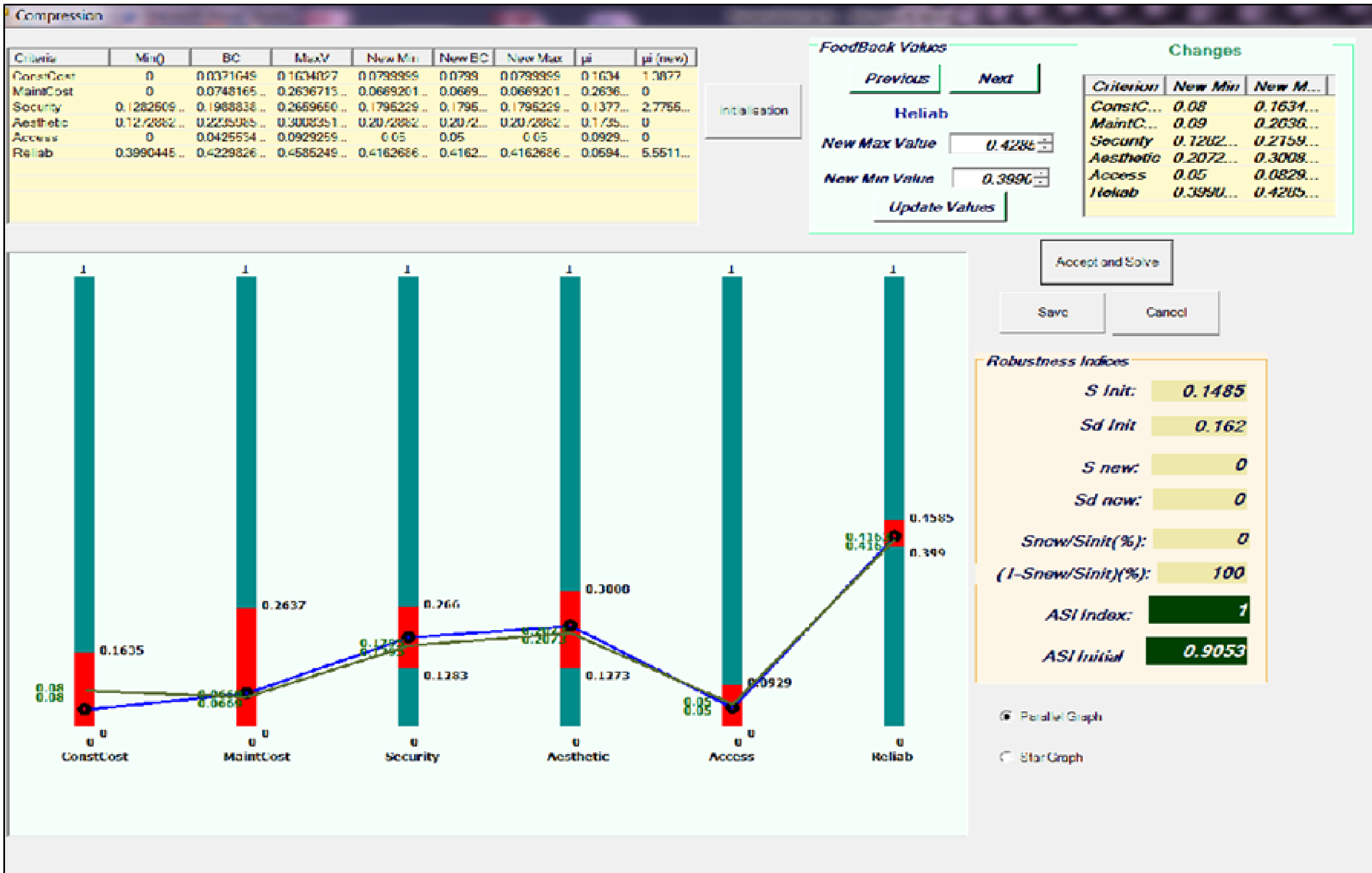
Feedbacks – Prioritisation of Criteria

Within this feedback process the DM is asked to provide further information concerning the priority of the criteria for a selected pair (g_i, g_j) **triggered from the existence of a priority rank reversal in the estimated hyper-polyhedron**.

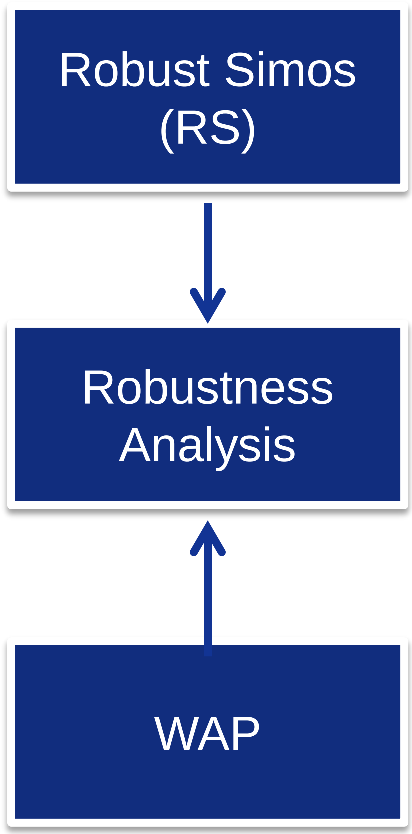


Feedbacks – Shrinking the hyper-polyhedron

The additional preference information concerning extreme points of the criteria weights **sets new lower and upper limits of the criteria weights** by inserting new constraints in the LPs of post-optimality analysis



Suite of RA Tools (3)



WAP vs Simos method

It enriches the preferential information used in a friendly and comprehensive by the DM way and at the same time it leads to the estimation of weighting vectors with higher robustness.

WAP Method	Simos Method
Prioritise Criteria (linear constraints)	Prioritise Criteria (using cards)
Increase z_r indices to increase weights difference	Insert white cards to increase weights difference
Solving LP to calculate weights	Calculate weights using Simos algorithm
Perform Robustness Analysis	NA
Continue with feedbacks if necessary	NA

WAP as a Robust Simos Method

The WAP method **includes several robustness rules** proposed by Siskos and Tsotsolas (2015) concerning the production of tangible and adequately supported results when Simos method is used. In particular the following rules are followed:

- Computation of the variation range of the weight of each one of the *n* criteria by solving 2*n* linear programs of Max-Min type (**rule 1**)
- Computation of the average weighting vector ("barycenter") of all different vectors (from the *2n* solutions obtained in the former rule), as a more representative weighting solution in the hyper-polyhedron (M-N Average) (**rule 2**)
- Visualisation of the ranges of variation of the criteria weights for a more comprehensively perceive the extent of the possible instability (**rule 5**)
- Computation of the robustness measure ASI (Average Stability Index) (**rule 7**)

Additionally to the aforementioned rules the WAP method includes a process for receiving extra information from the DM concerning the difference between the weights of the criteria (**feedbacks**).

WAP - z_r indices

The key point of the proposed approach is the use of the z_r indices for every pair of successive criteria or sets of ex aequo criteria sorted according to their ranking.

z_r: how many times a criterion is more important than the previous one in the ranking

DM is not asked to identify precisely these z_r indices but instead a range of value [$zmin_r$, $zmax_r$].

For two successive criteria or sets of ex aequo criteria, (ex. g_r, g_{r+1}) the range [$zmin_r$, $zmax_r$] is identified, so as:

$$zmin_r \leq z_r \leq zmax_r,$$

$p_r = z_r \, p_{r+1}$, where p_r is the weight of g_r and p_{r+1} is the weight of g_{r+1}.

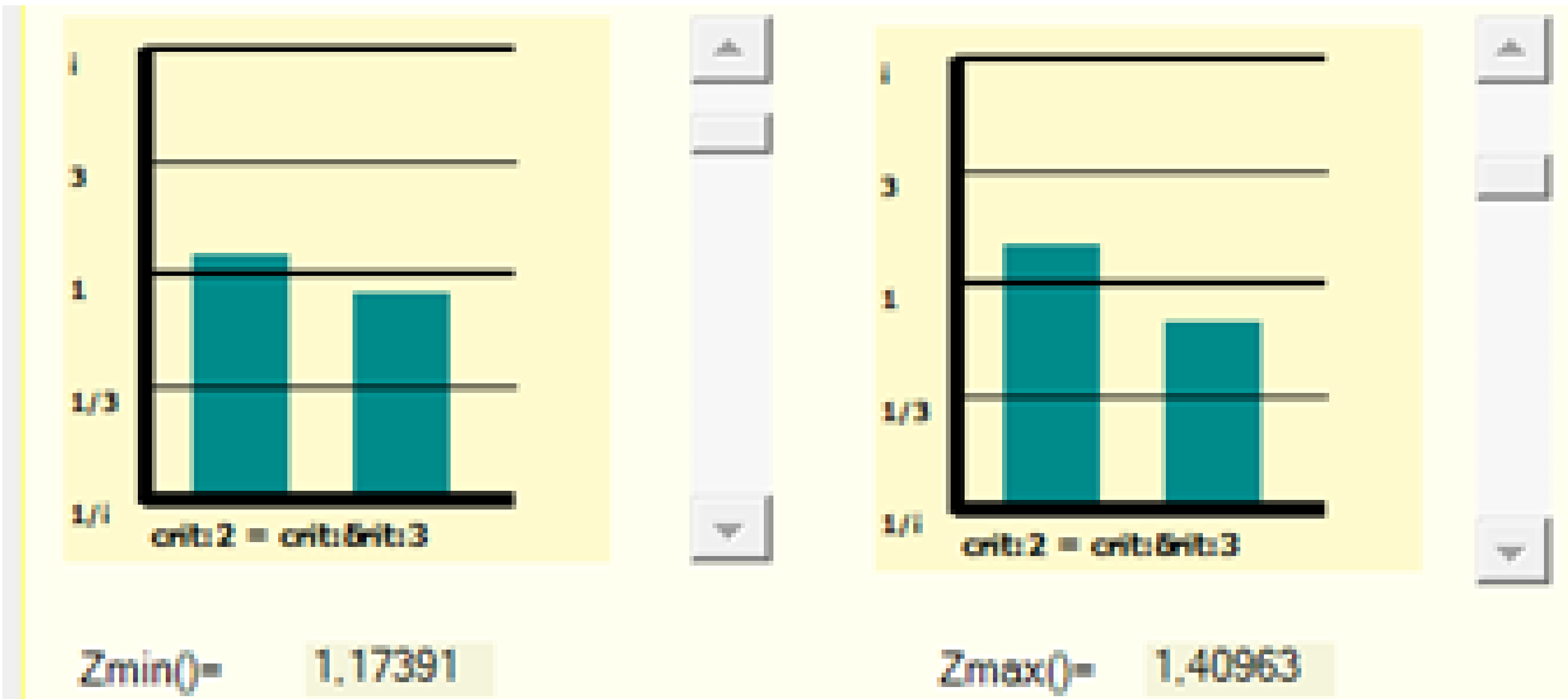
The z index, used in revised Simos, can be directly calculated by the product of the z_i indices:

$$z_1 \, z_2 \ldots z_{m-1} = (p_1/p_2)(p_2/p_3)..(p_{m-1}/p_m) = p_1/p_m = z.$$

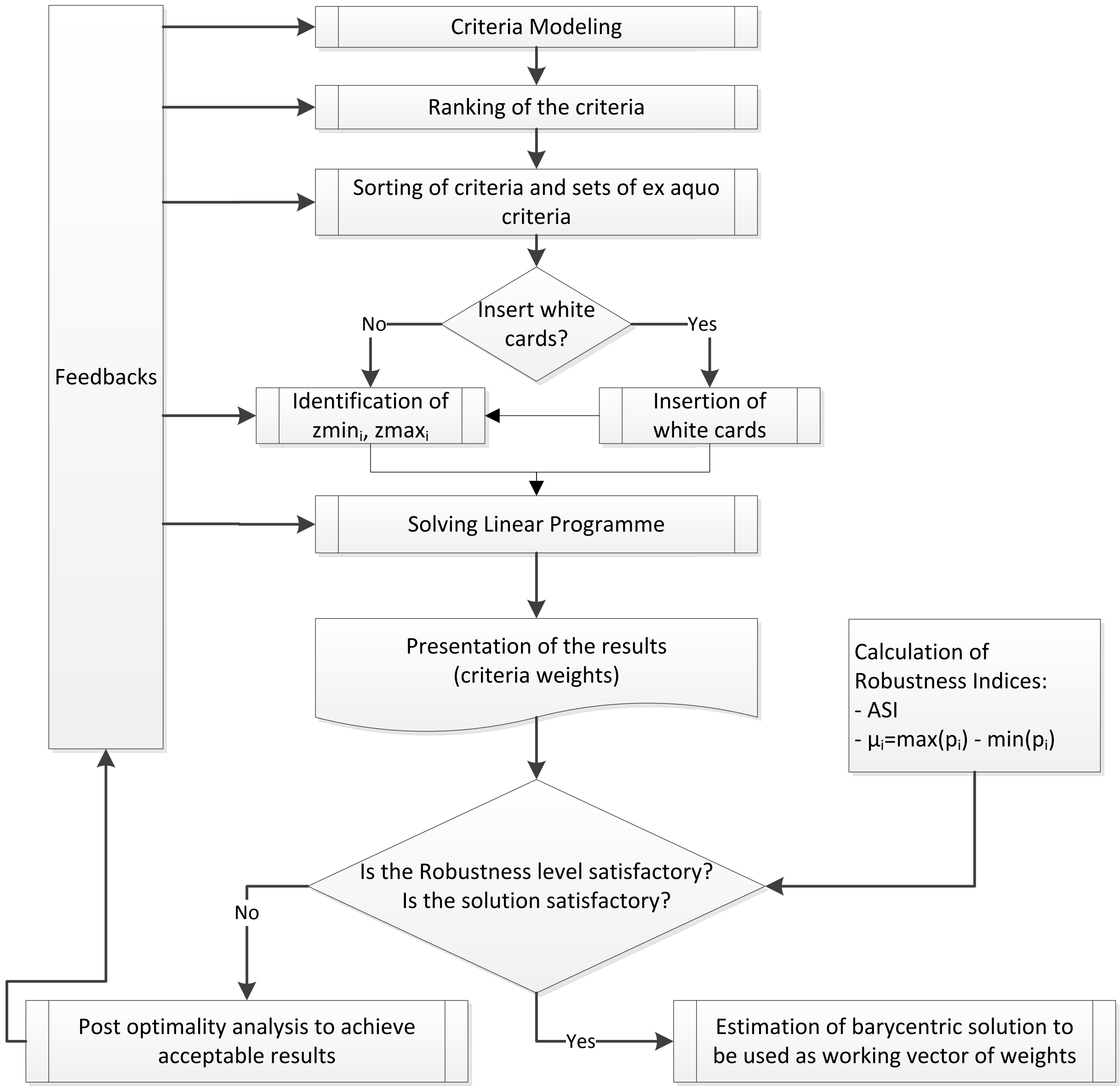
WAP - z_r indices in the WAP tool

In order to make the whole process easier for the DM to identify these values, special visual interactive techniques were developed and implemented in the WAP tool.

Through visual techniques the DM is asked to express the borders of the ranges of the relative importance between two consecutive criteria or set of ex aequo criteria for all of the pairs



WAP – Flow diagram



WAP - LP

Having identified the $zmin_r$ and $zmax_r$ for all the pairs of the successive classes The following L.P. is constructed and solved.

Min p_i & *Max* p_i, for $i = 1, 2, \ldots, n$

s.t.

$p_i - p_{i+1} = 0$, if g_{i+1} is followed by g_i, and g_{i+1} belongs to the same importance class (j) as g_i,

$w_h - p_j \geq \delta$ and $p_{j+1} - w_h \geq \delta$, if the DM inserts a white card w_h between criteria i and i+1. δ is a very small number, let's say 0.01

$p_i - p_{i+1} \geq zmin_j$, $p_i - p_{i+1} \leq zmax_j$, if g_r is followed by g_{i+1}, g_i belongs to a higher importance class j and g_{i+1} belongs to class j+1

$p_1 + p_2 + \ldots + p_n = 1$, (normalization rule)

$p_1 \geq 0, p_2 \geq 0, \ldots, p_n \geq 0$

WAP - Robustness Measures

$\mu_i = \left(\max\left(p_{ij}\right) - \min\left(p_{ij}\right)\right),$

p_{ij} the weight of the i criterion of the j, $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$, n: the number of criteria and m: the number of vertices of hyper-polyhedron.

$$ASI = 1 - \frac{\sum_{i=1}^{n} \sqrt{\left(m\left(\sum_{j=1}^{m} (p_i^j)^2\right) - \left(\sum_{j=1}^{m} p_i^j\right)^2\right)}}{m\sqrt{(n-1)}}$$

n: the number of criteria, m: the number of vertices of hyper-polyhedron

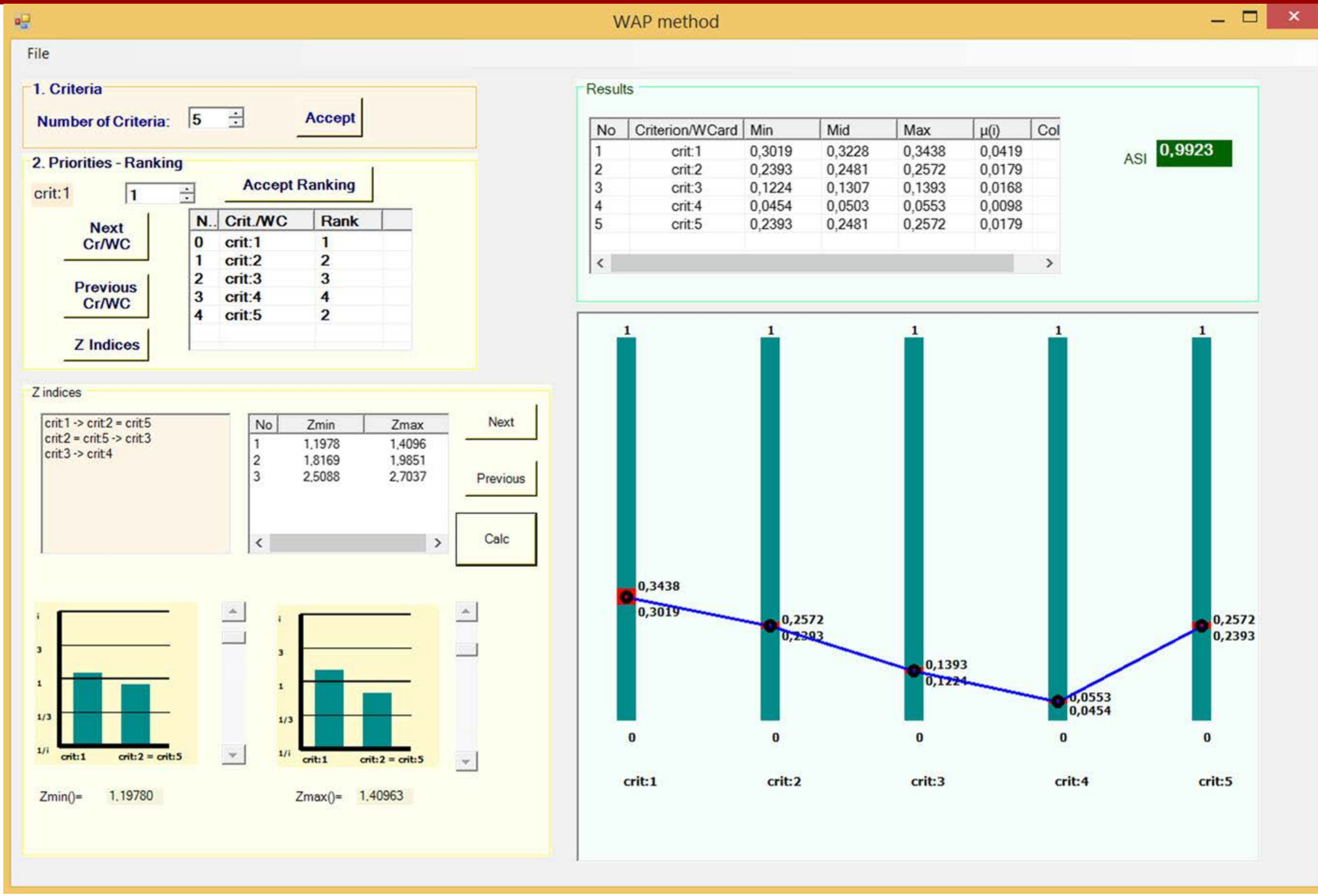
WAP - Example (1)

Ranking	Criteria
1	crit.1
2	crit.2, crit.5
3	crit.3
4	crit.4

Criterion	Min z_i	Max z_i
z_1: crit.1 P (crit.2, crit.5)	1.1978	1.4096
z_2: (crit.2, crit.5) P crit.3	1.8169	1.9851
z_3: crit.3 P crit.4	2.5088	2.7037

$$
\begin{aligned}
&Min\ p_i \ \&\ Max\ p_i,\ \text{for } i = 1, 2, \ldots, 5\\
&\qquad \text{s.t.}\\
&p_1 - p_2 \geq 1.1978,\ p_1 - p2 \leq 1.4096\\
&p_2 - p_5 = 0,\\
&p_5 - p_3 \geq 1.8169\ p_5 - p_3 \leq 1.9851\\
&p_3 - p_4 \geq 2.5088,\ p_3 - p_4 \leq 2.7037\\
&p_1 + p_2 + \cdots + p_5 = 1\\
&p_1 \geq 0,\ p_2 \geq 0,\ \ldots,\ p_5 \geq 0
\end{aligned}
$$

WAP - Example (2)



WAP - Example (3)

Criteria	Min	Max	Barycentric	μ_i
crit.1	0.3019	0.3438	**0.3228**	0.0419
crit.2	0.2393	0.2572	**0.2481**	0.0179
crit.3	0.1224	0.1393	**0.1307**	0.0168
crit.4	0.0454	0.0553	**0.0503**	0.0098
crit.5	0.2393	0.2572	**0.2481**	0.0179

WAP - Example (4)

The example, used for comparison purposes ,of Revised Simos and RSM includes the 5 criteria (named crit.1, crit.2, ..., crit.5) and 3 white cards (named Wc.1, Wc.2, Wc.3). The ranking of the criteria and white cards are:

No	Position of criteria (increasing importance)	Preferences
1	4	crit.1
2	3	crit.2, crit.5
3		Wc.1
4	2	crit.3
5		Wc.2
6		Wc.1
7	1	crit.4

WAP - Example (5)

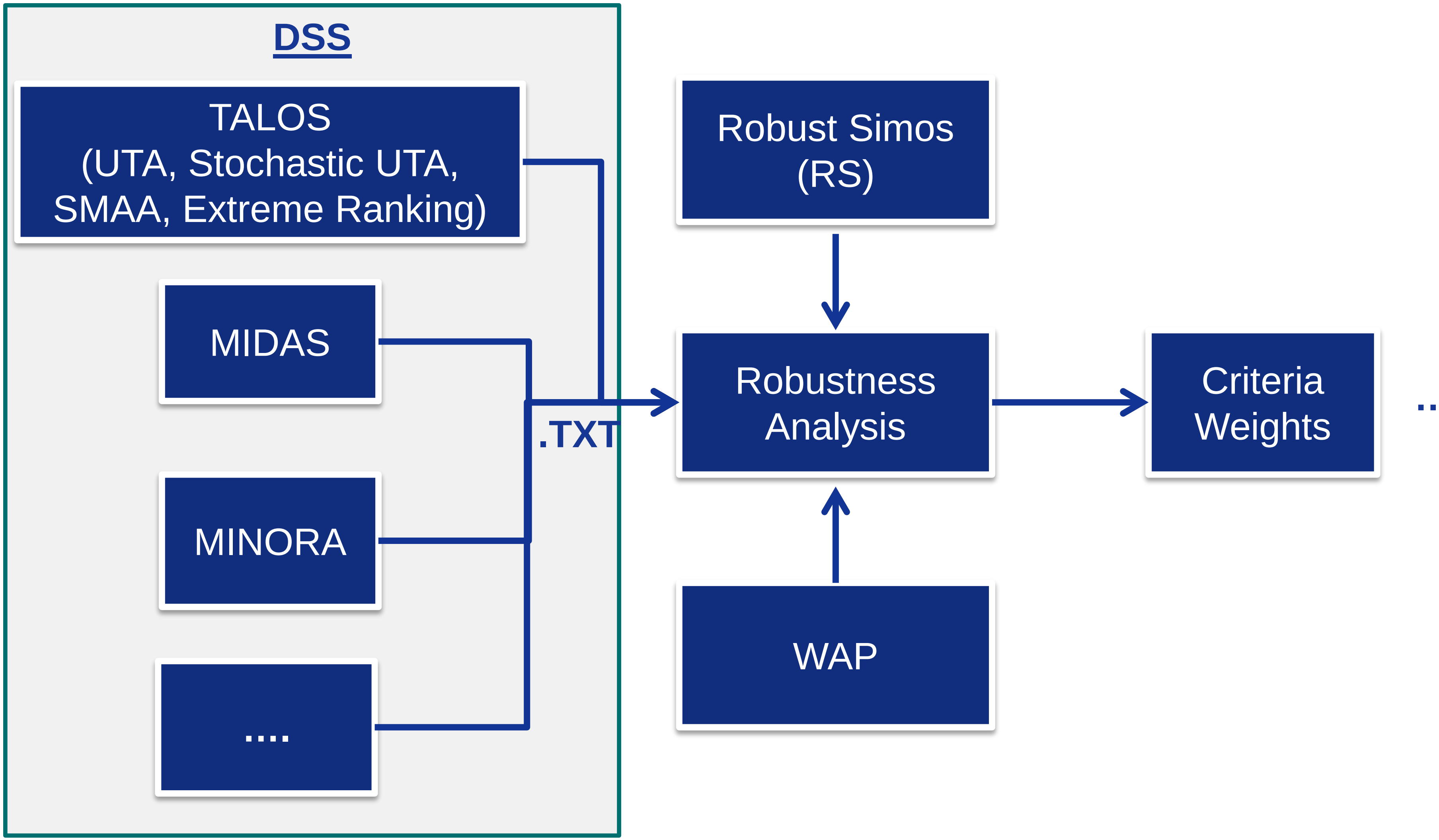
Criteria	Ranking	Revised Simos Weights	RSM weights			WAP method		
			Min	Max	BC	Min	Max	BC
crit.1	1	**0.290**	0.2789	0.3776	**0.3221**	0.3019	0.3438	**0.3228**
crit.2	2	**0.249**	0.2081	0.2633	**0.2363**	0.2393	0.2572	**0.2481**
crit.3	3	**0.167**	0.1351	0.1869	**0.1558**	0.1224	0.1393	**0.1307**
crit.4	4	**0.045**	0.0426	0.0581	**0.0496**	0.0454	0.0553	**0.0503**
crit.5	2	**0.249**	0.2081	0.2633	**0.2363**	0.2393	0.2572	**0.2481**

WAP - Example (6)

The newly estimated weighting vector by the WAP method, after feedbacks

WAP method			
Criteria	Min	Max	BC
crit.1	0.3019	0.3124	**0.3071**
crit.2	0.2505	0.2572	**0.2538**
crit.3	0.1283	0.1393	**0.1337**
crit.4	0.0476	0.0553	**0.0514**
crit.5	0.2505	0.2572	**0.2538**

Suite of RA Tools (4)



The need

Extensible:

- Support new processes or extend the existing ones
- Incorporate new algorithms
- Accept new forms of data schemas
- Produce new forms of data schemas

The need

Fast in development:

- Adopt a solid architecture scheme based on popular and fully supported standards
- Re-use code modules for repeated tasks
- Extended use of libraries for basic and common tasks
- Use a standard vocabulary

The need

Interoperable:

- Fully open and transparent to the external world
- Use well-accepted standards for the description of the modules – entities
- Use of a very well defined and extensible vocabulary
- Adopt a stable and open framework for modular development → **diviz**

The tools

A set of appropriate elements will be used to build this modular DSS:

- Elements (algorithms) from our already developed DSSs:
  - Stochastic UTA (TALOS)
  - MUSA-DSS
  - MIDAS
  - MINORA
  - …..

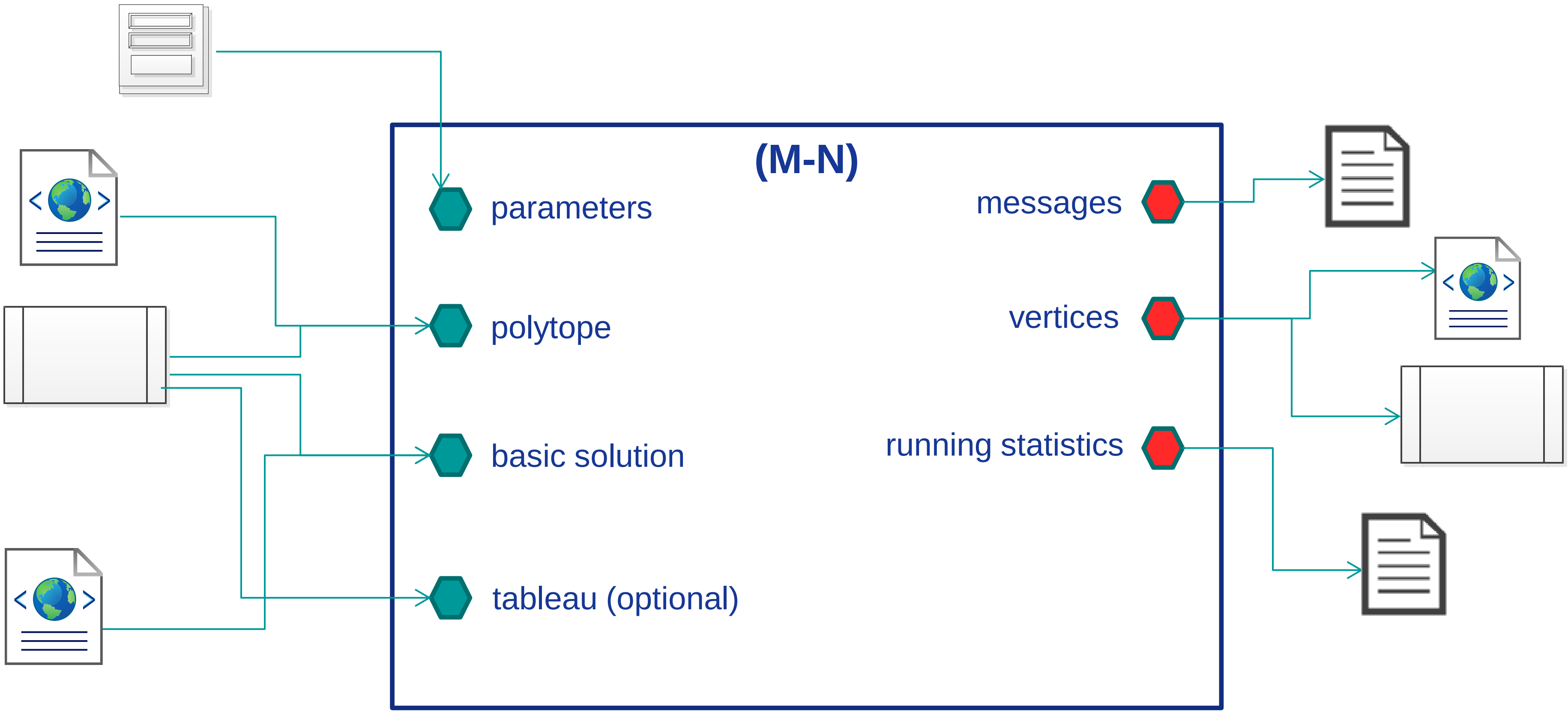
- XMCDA, a standardized XML vocabulary
- SOAP protocol for exchanging structured information – Web services
- Libraries (such as Kappalab box for capacity calculation and integral manipulation on a finite setting)

The architecture

Let’s see for example the transformation of an existing module found in two DSSs of our team :

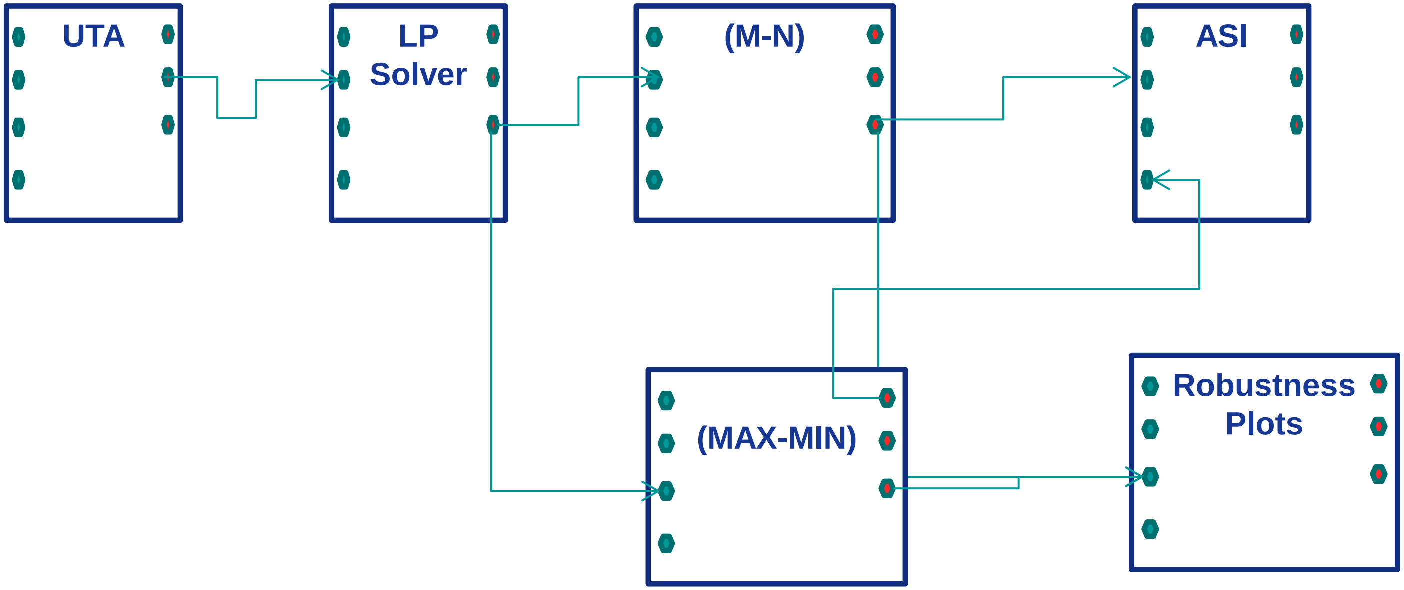
The architecture

The new module of the Manas- Nedoma Algorithm (M-N):



The architecture

How M-N module collaborates with other modules to implement a whole procedure:



Next Steps

1

Adjust the DSS shell based on XMCDA standards.

2

Transform the elements of the DSS into modules compatible with XMCDA.

Create web services

3

Add any necessary new modules – algorithms.

Questions